Linguagens Regulares

Lema do Bombeamento
Linguagens regulares e não-regulares

Lema do Bombeamento para LR

- Como decidir que uma linguagem não é regular?
  - Toda linguagem regular satisfaz o Lema do bombeamento (LB). Lemas são artefatos práticos, bons para provas.
  - Se alguém apresenta a você uma LR falsa, use o LB para mostrar a contradição, pois ela não vai satizfazer o LB.
- Veja que toda cadeia maior que o número de estados do AF força um estado do AF a se repetir…

- O lema do bombeamento (Pumping Lemma) nos diz que:
  - qualquer cadeia suficientemente longa z de uma LR pode ser decomposta em 3 partes:
  - z = uvw, de maneira que podemos construir outras cadeias da linguagem pela repetição da parte central v.

- Todas as cadeias da forma u v^i w são também da linguagem.
  - Ou seja, podemos acionar a *bomba* quantas vezes quisermos, para criar quantas sentenças novas da linguagem desejarmos: uw, uvvw, uvvvw, …

Mostrando que uma Linguagem não é Regular

- Para mostrar que uma linguagem não é regular, mostramos que
  - não há como decompor uma cadeia (qualquer, arbitrariamente longa) da linguagem de forma que seja possível *bombear* e continuar na linguagem.
  - Usamos ele para mostrar que CERTAS linguagens não são regulares. E são MUITAS para as quais aplicamos.

E para LR?

- Entretanto, o lema não dá uma condição suficiente para a linguagem ser regular,
  - pois para algumas não regulares o lema é verdadeiro.
  - Assim, o lema não pode ser usado para PROVAR que uma linguagem é regular, embora ele seja aplicável/verdadeiro para todas as LR´s.
    - Todavia, se o lema se aplica, nós podemos facilmente construir o AF que reconhece a linguagem, então dizemos que ela é regular.

Se L é uma linguagem regular, então:

existe uma constante natural n tal que,

para toda cadeia z de L com comprimento maior ou igual a n (*n é chamada "tamanho do bombeamento"*) , pode ser decomposta em três cadeias u, v, w

(z = uvw) de forma que

- **$|uv| \leq n$**
- **$v \neq \lambda$ (isto é, $|v| >= 1$) e**
- **para qualquer $i \geq 0$, $u\,v^i\,w \in L$ (veja que i=0 significa remover v).**

Observem a afirmação acima: existe uma

Lema em termos de algoritmo

Lemma.

For every **regular** language L,

there exists a number $p \geq 1$ such that

for every word $w \in L$ with at least p letters

there exist x, y, z with $w = xyz$ and $|y| > 0$ and $|xy| \leq p$ such that

for every number $i \geq 0$

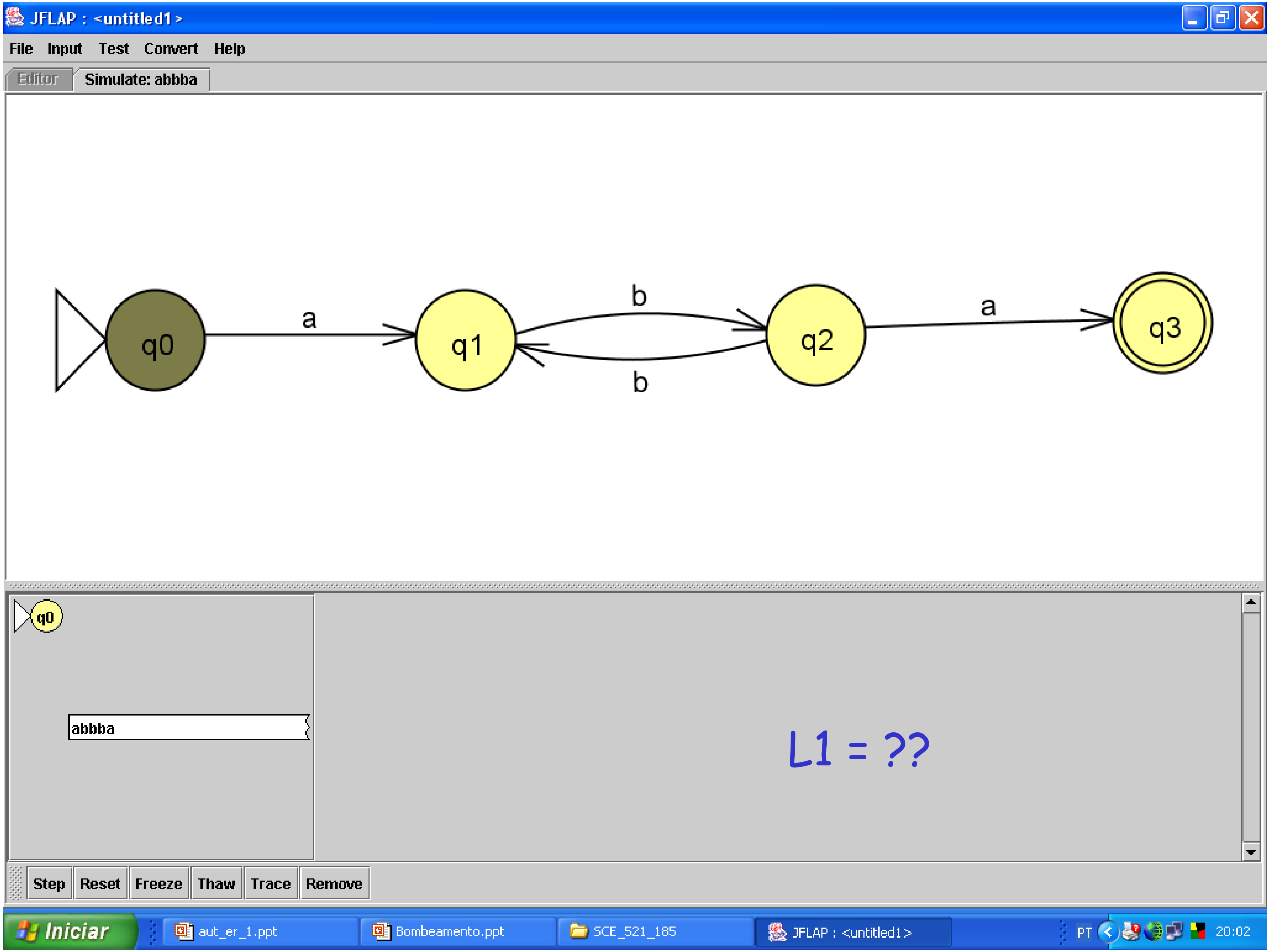
$xy^iz \in L$.

- **Demonstração (simplificada): Baseia-se no fato de que para as cadeias *longas* z é necessário usar pelo menos um loop de estados num AFD que aceite a linguagem.**

- **Assim, os símbolos de u são usados para chegarmos a um estado q do loop;**
- **os símbolos de v são usados para dar a volta no loop, de volta ao estado q;**
- **os símbolos de w são usados para ir de q até um estado final.**
- **Portanto, podemos dar quantas voltas no loop quisermos, e repetir v um número qualquer i de vezes: $u\,v^i\,w$.**

As cadeias curtas (comprimento < n) não são consideradas porque podem ser aceitas sem passar por nenhum loop.

Exemplo de demonstração do lema



- Pelo Lema do Bombeamento, temos que:

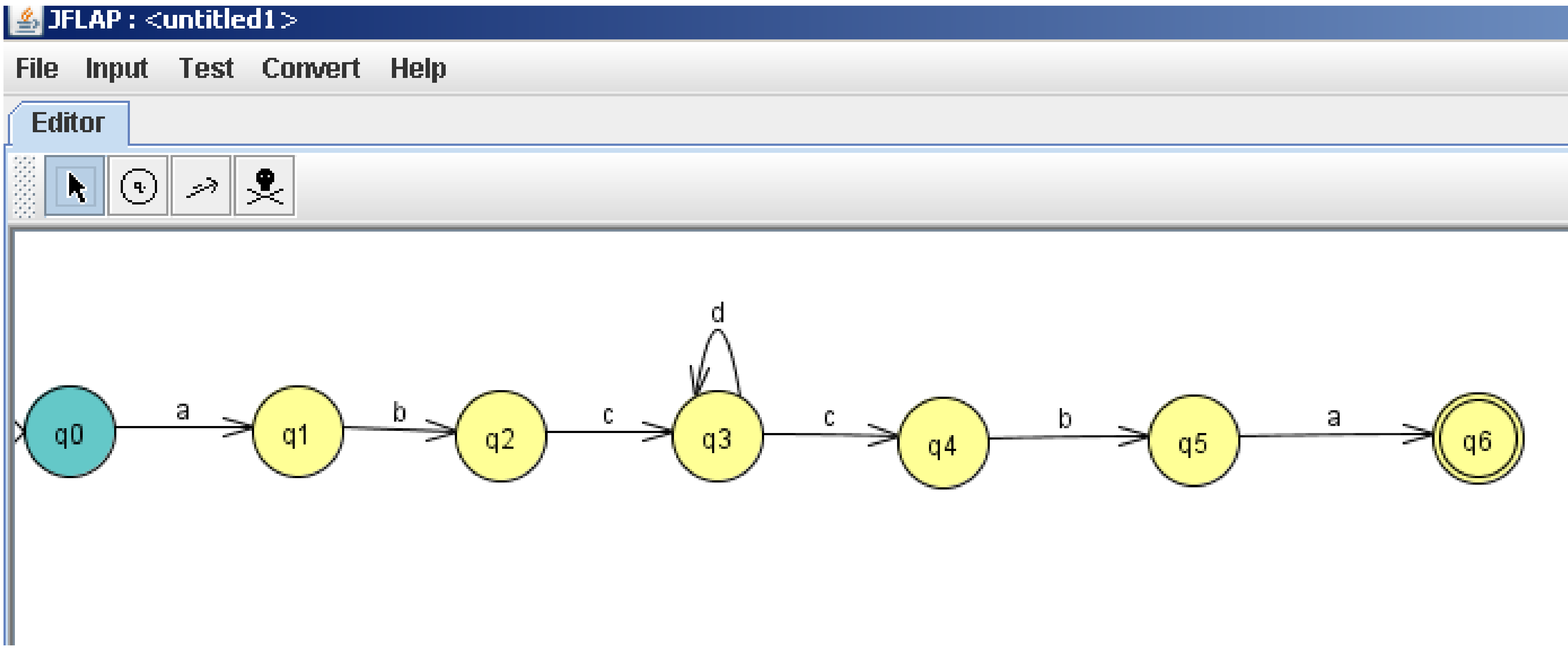
- n = 4 (depende da linguagem, pois depende do número de estados)
- Para z = abbba |z| = 5 >= 4
  - q = q1
  - u = a
  - v = bb |uv| <= 4; |v| >= 1
  - w = ba

Para todo i >= 0 a $(bb)^i$ ba $\in$ L1 pois L1 = $\{ab^n a \mid n >= 1$ e impar$\}$ ou $\{ab^{2n+1}a \mid n>=0\}$

Exercício

- Mostrem o lema se aplica para L2 = {abc d^n cba | n >= 0}, que é regular

AF para L2


JFLAP : <untitled1>
File
Input
Test
Convert
Help
Editor
d
q0
a
q1
b
q2
c
q3
c
q4
b
q5
a
q6

- Pelo Lema do Bombeamento, temos que:

- n = 7 (depende da linguagem, pois depende do número de estados)
- Para z = abcdcba |z| = 7 >= 7
  - q = q3
  - u = abc
  - v = d |uv| <= 7 ; |v| >= 1
  - w = cba

- (ou z = abcddcba; |z| =8 >=7; abc $(dd)^i$ cba $\in$ L2, i >=0)

- Para todo i >= 0 abc $(d)^i$ cba $\in$ L2 pois L2 = {abc d^ncba | n >=0}

Como escolhemos o n do lema?

- Pode ser usado o número de estados do AF que estamos trabalhando, para a demonstração de que o lema se aplica para uma LR.

- Observem que podemos ter vários AF para reconhecer uma dada LR.

Exemplo de linguagem não regular $\{uu^R v \mid u,v \quad \{0,1\}^+\}$ para a qual o lema é aplicável

- Esta linguagem pode ser bombeada com n = 4.
- Seja w = xyz
- Suponha que $w=uu^R v$ tem comprimento de pelo menos 4, por exemplo, 0011

- Se *u* tem tamanho 1, então $|v| \geq 2$ e *y* pode ser o primeiro caractere em *v*.
- x = 00; y = 1; z = 1
- |xy| <= 4 OK, pois 3 <= 4
- |y| <= 1 OK, pois |y| = 1
- $001^i\,1 \in L$ para qualquer i >= 0 OK, pois 001 є L; 0011 є L; 00111 є L e assim por diante.

Provando que L não é regular

$L = \{a^i b^i \mid i \geq 0\}$ (Já vimos que L é um LLC.)

Vamos mostrar, por contradição, que L não é regular.

Suponha que L é regular. Se L é regular, o Lema do Bombeamento se aplica, e existe n tal que a decomposição descrita pode ser realizada para qualquer cadeia de comprimento igual ou maior que n.

- Serão apresentadas 3 formas de prova, ou seja, 3 explicações

Prova 1

- $z = a^n b^n$
- $z = uvw$ onde:
- $|uv| <= n$ $|v| >=1$ e
- para todo $i>= 0$ uv^iw é palavra de L

O que é absurdo pois como $|uv| <= n$ uv é composta exclusivamente por a´s. Nesse caso, uv^2w não pertence a L pois não possui o mesmo número de a e b.

OU (Prova 2)

Seja $k = n + 1$.

Considere a cadeia $z = a^k b^k$. Qualquer decomposição $z = uvw$ deve ter em v o mesmo número de a´s e de b´s, para que a propriedade de que o número de a´s é igual ao de b´s se mantenha nas cadeias $u v^i w$.

- Se isso não acontecer, quando acrescentarmos mais um v (aumentando i de 1), obteremos uma cadeia fora da linguagem. Portanto, v deve ser da forma $a^j b^j$, com $j > 0$, já que v não pode ser vazia.
- Mas nesse caso, $u v^2 w$ conterá a cadeia $a^j b^j a^j b^j$, com pelo menos um a e depois um b, o que não pode acontecer na linguagem.

- Ou seja, nenhuma decomposição é possível, contrariando o Lema, e podemos concluir que L não é regular.

- OU (Prova 3)
- Considere a cadeira $z = 0^n1^n$; z pode ser dividida em 3 pedaços = uvw onde para i >= 0 a cadeia uv^iw pertence a L. Vamos considerar 3 casos para mostrar que este resultado é impossível:

- 1) a cadeia v consiste somente de 0´s. Neste caso, a cadeia uvvw tem mais 0´s do que 1 violando o LB.
- 2) a cadeia v consiste somente de 1´s. Este caso também dá uma contradição.
- 3) a cadeia v consiste de 0´s e 1´s. Neste caso a cadeia uvvw tem o mesmo número de 0´s e 1´s mas não estão na ordem desejada pois v = 0101. E assim ela não é membro de L o que é uma contradição.

- A contradição é inevitável se nós aceitamos a suposição de que B é regular. Assim, B não é regular.

Exercícios

Mostre que as linguagens abaixo não são regulares:

1) $L = \{0^n 1^{2n} \mid n >= 1\}$
2) $L = \{ xx^r \mid x \in \{0,1\}^*\}$
3) $L = \{0^n 1 0^n \mid n >= 1\}$
4) $L = \{0^n 1^m 2^n \mid$ n e m são inteiros quaisquer$\}$
5) $L = \{0^n 1^m \mid n <= m\}$

- Links úteis:
- http://www.cs.nuim.ie/~jpower/Courses/parsing/node19.html
- http://www.cse.msu.edu/~torng/460/Homework/sphw10.html
- www.eas.asu.edu/~cse355fa/cse355a-spring04/class-notes/fritz2-26.doc

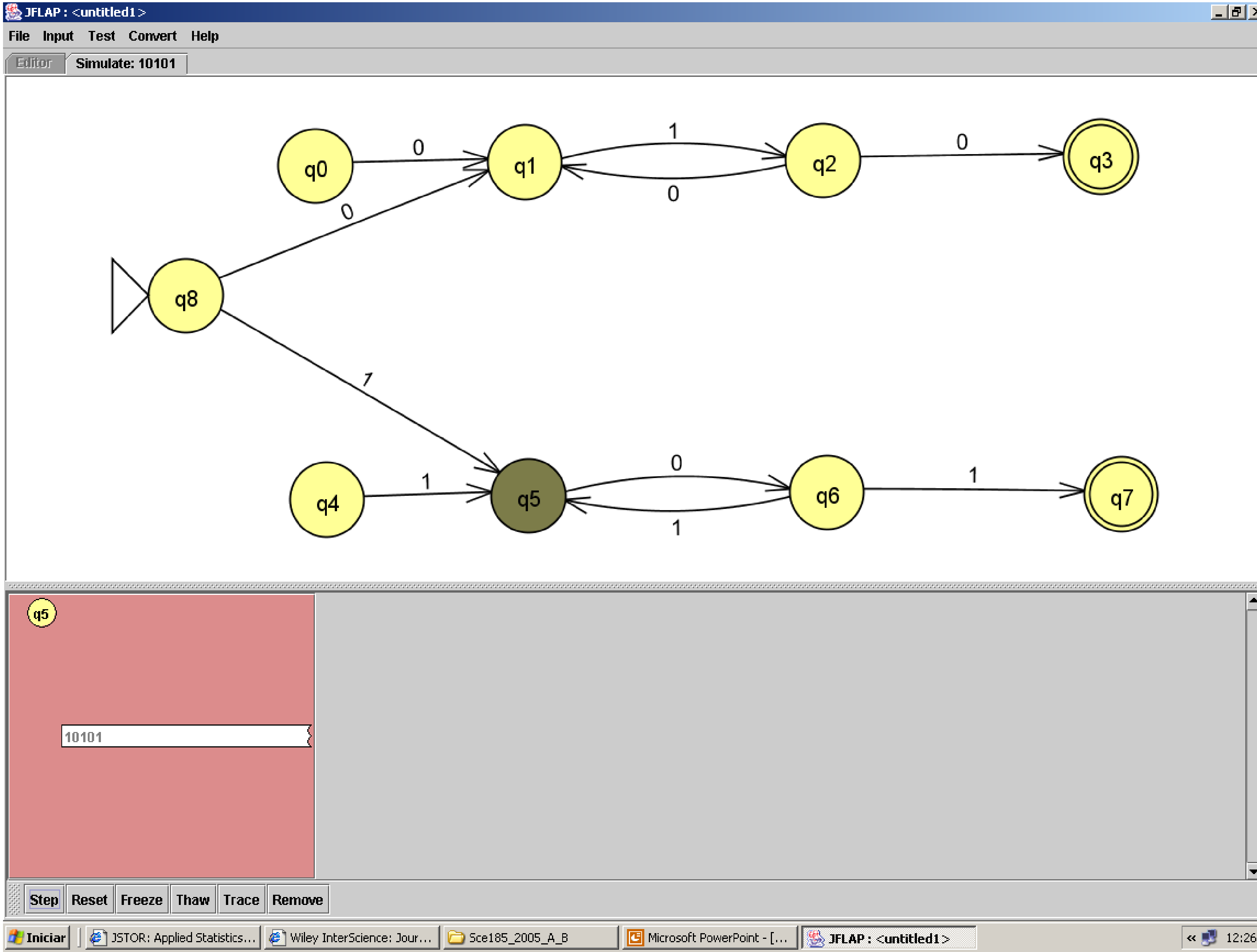
Mostre que a Linguagem abaixo satisfaz o Lema

- L = {w | w tem um número igual de ocorrências de subcadeias 01 e 10 e somente elas}.

- Por exemplo, 101 $\in$ a L pois contém 1 ocorrência de 01 e uma de 10 MAS 1010 não pertence pois contém dois 10 e um 01.
- 010 também pertence e 0101 não.
- Cadeias que pertencem a linguagem são:
  - 101, 10101, 1010101, ...,
  - 010, 01010,0101010,....

- Embora possa parecer que a máquina precise contar as cadeias acima como na linguagem 0^n1^n, podemos estar errados...

Uma das soluções é a união de 2 AF´s



Pelo Lema do Bombeamento, temos que:

n = 4

- Para z = 01010 |z| = 5 >= 4
  - q = q1
  - u = 0
  - v = 10 |uv| <= 4 ; |v| >= 1
  - w = 10
  - Para todo i >= 0 $(10)^i$ 10 $\in$ L

- Para z = 10101 |z| = 5 >= 4
  - q = q5
  - u = 1
  - v = 01 |uv| <= 4 ; |v| >= 1
  - w = 01
  - Para todo i >= 1 $(01)^i$ 01 $\in$ L

Jflap

- Verifiquem como o JFLAP trata o lema do bombeamento:
  - Um jogo de adversários (humano, máquina)
  - Utilizem o jogo para exercitar a prova de que as várias linguagens lá disponíveis não são LR.